We thank the reviewers for their overall positive feedback.

The main concern raised by **Reviewer #2** is that Assumption 3, which requires that the positive examples are sampled within a margin γ from the boundaries of the set K_n, makes the overall results too weak. Reviewer #2 rightly noted that for the distributions presented in the paper, γ should decay exponentially with n, and this may seem to be a strong requirement. While this is a valid concern, we stress that we did not use this assumption at all for the results on failure of gradient-descent. Since having a margin γ can only help the optimization, dropping this assumption simply makes the optimization harder, hence these results still hold. Similarly, other negative results in the paper, namely - the inability of shallow networks to express fractal distributions, hold without Assumption 3: this assumption only makes the approximation problem easier.

In fact, the only results that rely on Assumption 3 is Theorem 1 and its corollaries, which give positive results, stating that fractal distributions can be efficiently expressed by deep networks. While the existence of a margin simplifies the construction made in the proof of this theorem, we can prove this theorem even **without** Assumption 3. We give a sketch of such proof below. **To summarize, in order to answer the concern of Reveiwer #2 we will completely remove Assumption 3 from the final version, and adjust all the theorems accordingly.**

Following the suggestion of **Reviewer #3**, we ran an experiment on the Vicsek distribution of depth 6, where the examples are concentrated on the "fine" details of the fractal. Such distribution is hard to approximate by a shallow network, as shown in our theoretical analysis. We trained networks of various depth and width on this distribution (as in the experiments described in the original submission). The results are shown in Figure 1. As could be seen clearly, unlike distributions with "coarse" approximation curve (shown in the original submission), in this case the benefit of depth is not noticeable, and all architectures achieve an accuracy of slightly more than 0.5 (i.e., chance level performance).

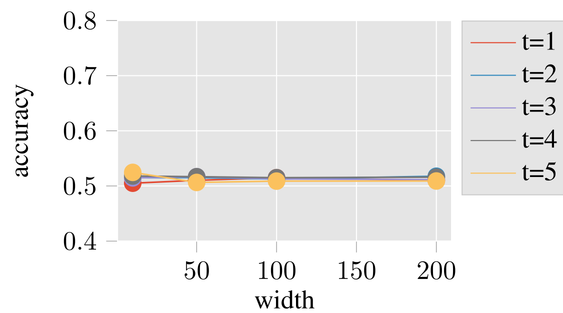

Figure 1: Performance on the "fine" Vicsek distribution.

We will additionally fix other minor issues raised by the reviewers in the final version.

Proof Sketch of Theorem 1 without Assumption 3

Lemma 2 *(without Assumption 3, standard construction of a ReLU network) There exists a neural-network with two hidden-layers such that $\mathcal{N}_{\mathbf{W},\boldsymbol{B}}(\boldsymbol{x}) < 0$ for $\boldsymbol{x} \notin [0,1]^d$, and $\mathcal{N}_{\mathbf{W},\boldsymbol{B}}(\boldsymbol{x}) \geq 0$ for $\boldsymbol{x} \in [0,1]^d$.*

Lemma 3 *(without Assumption 3) There exists a neural-network of width $\max\{dr, 3d\}$ with two hidden-layers ($k = 3dr, t = 3$) such that for any n we have: $\mathcal{N}_{\mathbf{W},\boldsymbol{B}}(K_n) \subseteq K_{n-1}$ and $\mathcal{N}_{\mathbf{W},\boldsymbol{B}}(K_1 \setminus K_n) \subseteq \mathcal{X} \setminus K_{n-1}$*

Proof Simple modification to the proof of Lemma 3 in the original submission. ■

Lemma 4 *(without Assumption 3) There exists a neural-network of width $2dr$ with two hidden-layers ($k = 2dr, t = 3$) such that for any n we have: $\mathcal{N}_{\mathbf{W},\boldsymbol{B}}(\mathcal{X} \setminus K_1) < 0$ and $\mathcal{N}_{\mathbf{W},\boldsymbol{B}}(K_1) \geq 0$.*

Proof Using Lemma 2, and following the same proof of Lemma 4 in the original submission. ■

Proof of Theorem 1 (without Assumption 3). We follow a proof similar to the one given in the original submission. Instead of the original definition of h, we define $h(\boldsymbol{x}) = [g(\boldsymbol{x}_{1\ldots d}), \boldsymbol{x}_{d+1} - \sigma(\boldsymbol{x}_{d+1} - \tilde{g}(\boldsymbol{x}_{1\ldots d}))]$. Then, constructing H as in the original proof satisfies that $H(\boldsymbol{x})_{d+1} < 0$ if and only if $x \notin K_n$: if the $d+1$ coordinate of some layer becomes negative, it stays negative throughout the network (since the $d+1$ coordinate of each layer is just the minimum of the $d+1$ coordinates of previous layers). Therefore, a network that outputs $H(\boldsymbol{x})_{d+1}$ achieves the required. ■